

Development of Android Application for Multi-user Authentication

Hubert Lin, Yuhan Hu

Introduction

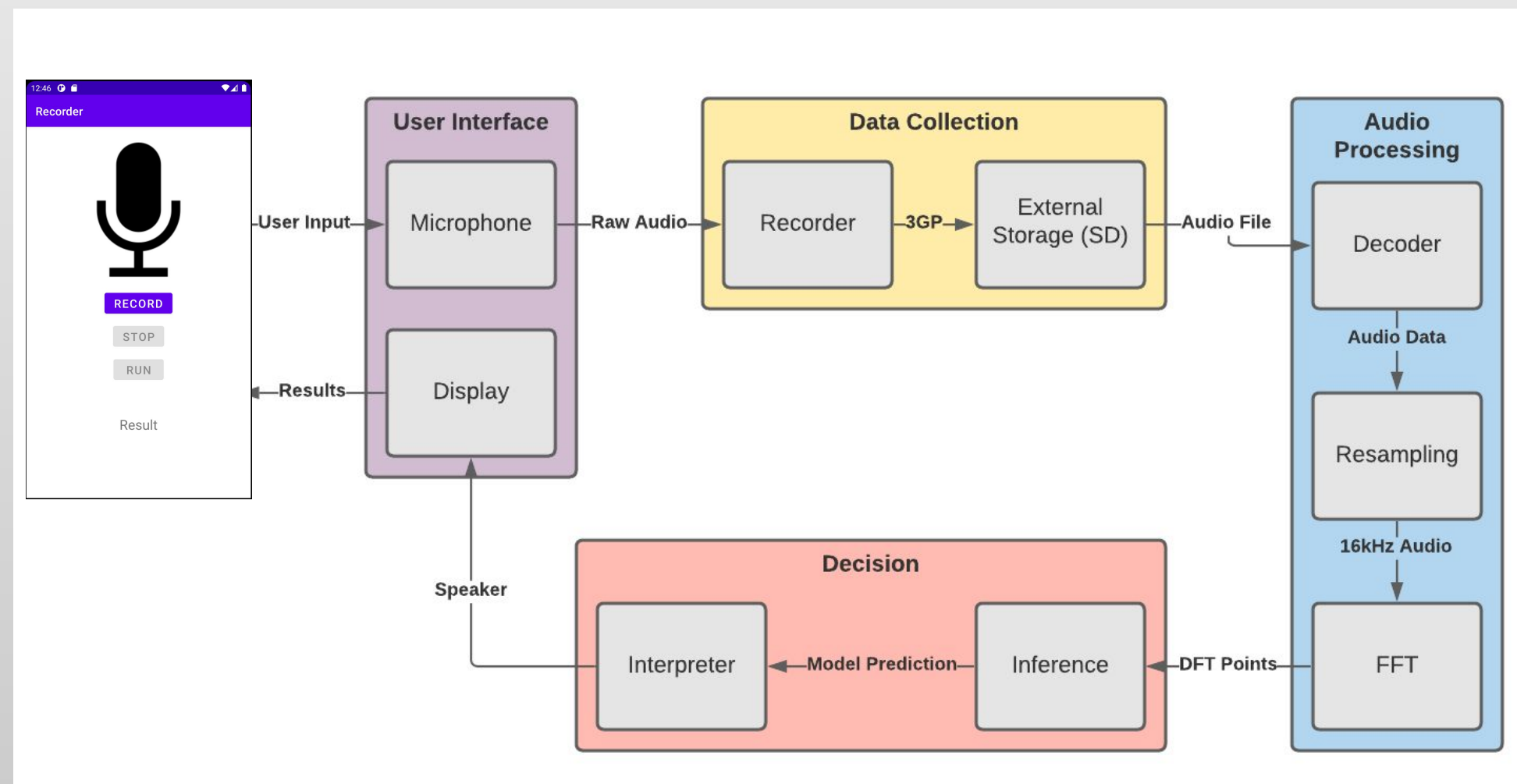
- Multi-speaker identification is useful for determining who is speaking in situations where more than one speaker may be present
- Wide range of applications including smart device voice authentication, voice transcription, and speaker tracking
- Project focus on development of a mobile application that can identify a specific user from a list of multiple users based on their voice data
- Android provides basic support for loading and using machine learning models in applications
- Platform choice leverages large number of smartphone users, existing hardware, and ease of accessibility

Methodology

- Build, train, and evaluate a 1D convolutional neural network model in Python using Tensorflow
- Create mobile application to record user voice data using Android Studio
- Evaluate trained model on a physical device using test application
- Write audio processing code to prepare raw data for model inference
- Integrate all components into single application interface

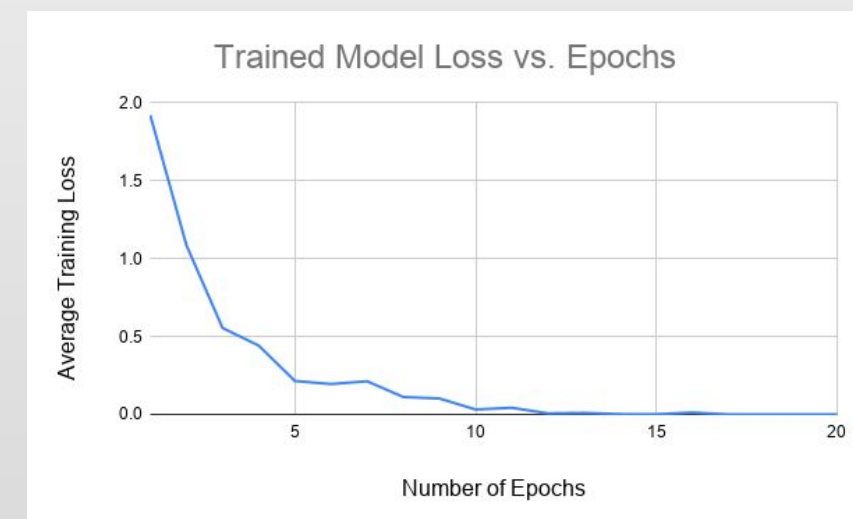
System Modules

- **User Interface:** Allows users to interact with the application by enabling user input and displaying results
- **Data Collection:** A recorder object captures and stores raw audio data to a file for additional processing
- **Audio Processing:** Decodes, transforms, and reformats data to match size expected by the model
- **Decision:** A pretrained model takes the processed input and produces probability values used to predict the most likely speaker



Results

- Model trained using user voice recordings
- All modules functional and application fully integrated
- Current product predicts identity of speaker when tested on a physical device



Conclusion

- We were able to build and train a machine learning model using our own data
- Gained familiarity with Android Studio and created a working application
- Future work can be done to accommodate for more users, add additional features, and improve functionality